

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-287990

(43)Date of publication of application : 04.10.2002

(51)Int.Cl.

G06F 9/54
G06F 9/445

(21)Application number : 2001-087577

(71)Applicant : CANON INC

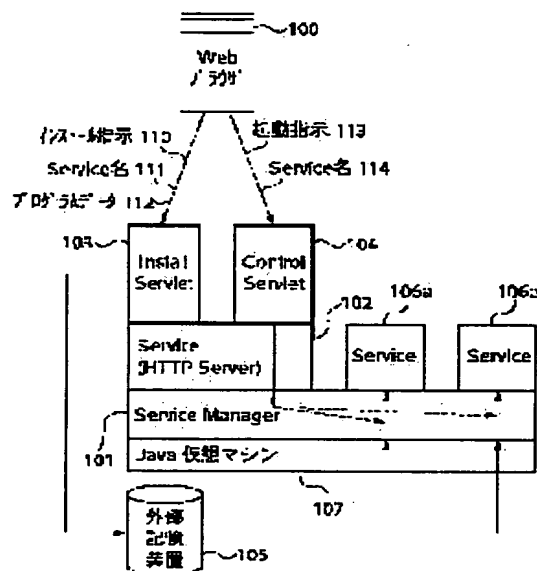
(22)Date of filing : 26.03.2001

(72)Inventor : FUJISAWA KUNIMASA

(54) APPLICATION MANAGING DEVICE, APPLICATION MANAGING METHOD, RECORDING MEDIUM, AND PROGRAM**(57)Abstract:**

PROBLEM TO BE SOLVED: To provide an application managing device, an application managing method, a recording medium, and a program, capable of installing a program in built-in apparatus, and starting the installed program, after shipping the built-in apparatus such as a copy machine, a printer, a multifunction printer(MFP) having a copy function/a printer function/a facsimile function, and a digital camera.

SOLUTION: When a name 111 of a Service to be installed and program data 112 of the Service are transferred as a program installing instruction from a Web browser 110 to an InstallServlet 103 by an HTTP protocol, the InstallServlet 103 saves the program data 112 in an external memory 105 as a file, and registers a pass saving the program and the Service name 111 in a class path managing portion 202.



BEST AVAILABLE COPY

LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2002-287990
(P2002-287990A)

(43) 公開日 平成14年10月4日 (2002. 10. 4)

(51) Int.Cl.⁷

G 0 6 F 9/54
9/445

識別記号

F I

G 0 6 F 9/06

テームコード* (参考)

6 4 0 C 5 B 0 7 6
6 1 0 L

審査請求 未請求 請求項の数40 O L (全 21 頁)

(21) 出願番号 特願2001-87577(P2001-87577)

(22) 出願日 平成13年3月26日 (2001. 3. 26)

(71) 出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72) 発明者 藤澤 邦匡

東京都大田区下丸子3丁目30番2号 キヤ
ノン株式会社内

(74) 代理人 100081880

弁理士 渡部 敏彦

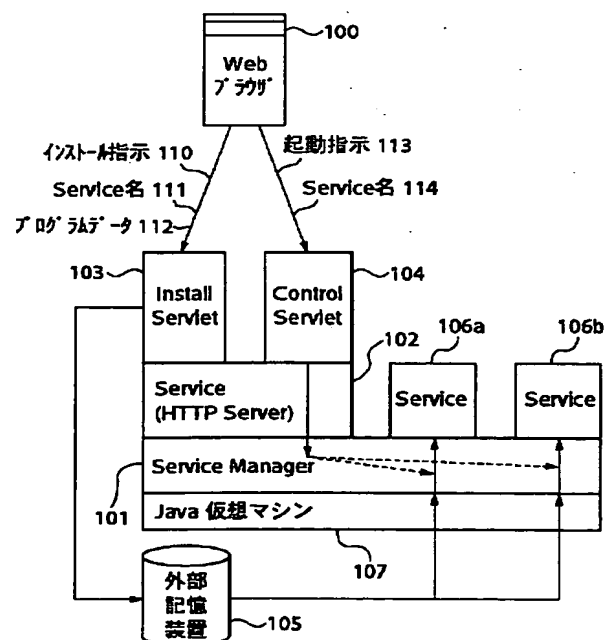
Fターム(参考) 5B076 AA07 AB10 BB06

(54) 【発明の名称】 アプリケーション管理装置、アプリケーション管理方法、記憶媒体及びプログラム

(57) 【要約】

【課題】 複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機 (MFP)、デジタルカメラ等の組込み機器を出荷した後に、組込み機器にプログラムをインストールしたり、更にインストールしたプログラムを起動したりすることを可能としたアプリケーション管理装置、アプリケーション管理方法、記憶媒体及びプログラムを提供する。

【解決手段】 Webブラウザ100からInstallServlet103に対し、HTTPプロトコルでプログラムのインストール指示110として、インストールすべきServiceの名称111、Serviceのプログラムデータ112が転送されてくると、InstallServlet103は、プログラムデータ112を外部記憶装置105にファイルとして保存し、更にプログラムを保存したパスとService名称111をクラスパス管理部202に登録する。



【特許請求の範囲】

【請求項1】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置であって、

アプリケーションのプログラムファイルに関する情報を記憶する記憶手段と、

クライアントと通信を行うネットワーク接続手段と、
前記クライアントから前記ネットワーク接続手段を介してインストール指示とプログラムファイルが送信されてきた場合、前記プログラムファイルを外部記憶装置に保存し、前記プログラムファイルの位置情報を前記記憶手段に登録する登録制御手段とを有することを特徴とするアプリケーション管理装置。

【請求項2】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置であって、

アプリケーションのクラスローダに登録する登録手段と、

アプリケーションのプログラムファイルの保存場所を記憶する記憶手段と、

クライアントと通信を行うネットワーク接続手段と、
前記クライアントから前記ネットワーク接続手段を介してアプリケーションの停止指示があった場合は、前記停止指示があったアプリケーションを停止し、前記クライアントから前記ネットワーク接続手段を介してアプリケーションの起動指示があった場合で、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成し、アプリケーションを起動し、前記作成したクラスローダを前記登録手段に登録し、前記作成したクラスローダによりクラスをロードし、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されている場合は、前記登録手段に登録されているクラスローダを取得し、前記取得したクラスローダによりクラスをロードし、アプリケーションを起動する起動・停止手段とを有することを特徴とするアプリケーション管理装置。

【請求項3】 アプリケーションの実行状態を管理する状態管理手段を有し、アプリケーションの実行に必要なメモリが不足した場合、前記状態管理手段に停止状態と記憶されているアプリケーションのクラスローダを前記登録手段から削除することを特徴とする請求項2記載のアプリケーション管理装置。

【請求項4】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置であって、
アプリケーションのクラスローダに登録する登録手段と、
アプリケーションのプログラムファイルの保存場所を記憶する記憶手段と、

クライアントと通信を行うネットワーク接続手段と、

前記クライアントから前記ネットワーク接続手段を介してアンインストール指示とアンインストール対象のアプリケーション名が送信されてきた場合、プログラムファイルが保存された外部記憶装置から前記アプリケーション名に対応するプログラムファイルを削除し、前記登録手段及び前記記憶手段から前記アプリケーション名に対応するエントリを削除する削除制御手段とを有することを特徴とするアプリケーション管理装置。

【請求項5】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置であって、

アプリケーションのプログラムファイルに関する情報を記憶する記憶手段と、

クライアントと通信を行うネットワーク接続手段と、

アプリケーション管理装置上で動作するアプリケーションの一つであるインストールアプリケーションとして動作し、前記クライアントから前記ネットワーク接続手段を介してプログラムファイルが送信されてきた場合、前記プログラムファイルを外部記憶装置に保存し、前記プログラムファイルの位置情報を前記記憶手段に登録する登録制御手段とを有することを特徴とするアプリケーション管理装置。

【請求項6】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置であって、

アプリケーションの起動・停止を行う起動・停止手段と、

アプリケーションのクラスローダに登録する登録手段と、

アプリケーションのプログラムファイルの保存場所を記憶する記憶手段と、

クライアントと通信を行うネットワーク接続手段と、

アプリケーション管理装置上で動作するアプリケーションの一つである起動・停止アプリケーションとして動作し、前記クライアントから前記ネットワーク接続手段を介してアプリケーションの停止指示があった場合は、前記停止指示があったアプリケーションを停止するように前記起動・停止手段に指示し、前記クライアントから前記ネットワーク接続手段を介してアプリケーションの起動指示があった場合で、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成し、アプリケーションを起動し、前記作成したクラスローダを前記登録手段に登録し、前記作成したクラスローダによりクラスをロードし、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されている場合は、前記登録手段に登録されているクラスローダを取得し、前記取得したクラスローダによりクラスをロードし、アプリケーションを起動するように前

記起動・停止手段に指示する起動・停止制御手段とを有することを特徴とするアプリケーション管理装置。

【請求項 7】 アプリケーションの実行状態を管理する状態管理手段を有し、アプリケーションの実行に必要なメモリが不足した場合、前記状態管理手段に停止状態と記憶されているアプリケーションのクラスローダを前記登録手段から削除することを特徴とする請求項 6 記載のアプリケーション管理装置。

【請求項 8】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置であって、

アプリケーションの起動・停止を行う起動・停止手段と、

アプリケーションのクラスローダを登録する登録手段と、

アプリケーションのプログラムファイルの保存場所を記憶する記憶手段と、

クライアントと通信を行うネットワーク接続手段と、

アプリケーション管理装置上で動作するアプリケーションの一つであるアンインストールアプリケーションとして動作し、前記クライアントから前記ネットワーク接続手段を介してアンインストール指示とアンインストール対象のアプリケーション名が送信されてきた場合、プログラムファイルが保存された外部記憶装置から前記アプリケーション名に対応するプログラムファイルを削除し、前記登録手段及び前記記憶手段から前記アプリケーション名に対応するエントリを削除する削除制御手段とを有することを特徴とするアプリケーション管理装置。

【請求項 9】 前記仮想マシンは、Java (Sun Microsystems の登録商標) 仮想マシンを含み、複数のアプリケーションが動作可能であることを特徴とする請求項 1、2、4、5、6、8 の何れかに記載のアプリケーション管理装置。

【請求項 10】 複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機、デジタルカメラ等の組込み機器に対するアプリケーションのインストール、起動、停止、アンインストールに適用可能であることを特徴とする請求項 1 乃至 8 の何れかに記載のアプリケーション管理装置。

【請求項 11】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法であって、クライアントからネットワークを介して前記機器にインストール指示とプログラムファイルが送信されてきた場合、前記プログラムファイルを外部記憶装置に保存し、前記プログラムファイルの位置情報を、アプリケーションのプログラムファイルに関する情報が記憶される記憶手段に登録することを特徴とするアプリケーション管理方法。

【請求項 12】 機器における仮想マシン上で動作する

アプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法であって、クライアントからネットワークを介して前記機器にアプリケーションの停止指示があった場合は、前記停止指示があったアプリケーションを停止し、前記クライアントから前記ネットワークを介してアプリケーションの起動指示があった場合で、前記起動指示があったアプリケーション用のクラスローダが、アプリケーションのクラスローダに登録される登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成し、アプリケーションを起動し、前記作成したクラスローダを前記登録手段に登録し、前記作成したクラスローダによりクラスをロードし、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されている場合は、前記登録手段に登録されているクラスローダを取得し、前記取得したクラスローダによりクラスをロードし、アプリケーションを起動することを特徴とするアプリケーション管理方法。

【請求項 13】 アプリケーションの実行に必要なメモリが不足した場合、アプリケーションの実行状態を管理する状態管理手段に停止状態と記憶されているアプリケーションのクラスローダを前記登録手段から削除することを特徴とする請求項 12 記載のアプリケーション管理方法。

【請求項 14】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法であって、クライアントからネットワークを介して前記機器にアンインストール指示とアンインストール対象のアプリケーション名が送信されてきた場合、プログラムファイルが保存された外部記憶装置から前記アプリケーション名に対応するプログラムファイルを削除し、アプリケーションのクラスローダに登録される登録手段、及びアプリケーションのプログラムファイルの保存場所が記憶される記憶手段から前記アプリケーション名に対応するエントリを削除することを特徴とするアプリケーション管理方法。

【請求項 15】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法であって、前記アプリケーション管理装置は、アプリケーション管理プログラム上で動作するアプリケーションの一つであるインストールアプリケーションとして動作し、クライアントからネットワークを介して前記機器にプログラムファイルが送信されてきた場合、前記プログラムファイルを外部記憶装置に保存し、前記プログラムファイルの位置情報を、アプリケーションのプログラムファイルに関する情報が記憶される記憶手段に登録することを特徴とするアプリケーション管理方法。

【請求項 16】 機器における仮想マシン上で動作する

アプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法であって、前記アプリケーション管理装置は、アプリケーション管理プログラム上で動作するアプリケーションの一つである起動・停止アプリケーションとして動作し、クライアントからネットワークを介して前記機器にアプリケーションの停止指示があった場合は、前記停止指示があったアプリケーションを停止するように、アプリケーションの起動・停止を行う起動・停止手段に指示し、前記クライアントから前記ネットワークを介してアプリケーションの起動指示があった場合で、前記起動指示があったアプリケーション用のクラスローダが、アプリケーションのクラスローダが登録される登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成し、アプリケーションを起動し、前記作成したクラスローダを前記登録手段に登録し、前記作成したクラスローダによりクラスをロードし、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されている場合は、前記登録手段に登録されているクラスローダを取得し、前記取得したクラスローダによりクラスをロードし、アプリケーションを起動するように前記起動・停止手段に指示することを特徴とするアプリケーション管理方法。

【請求項17】 アプリケーションの実行に必要なメモリが不足した場合、アプリケーションの実行状態を管理する状態管理手段に停止状態と記憶されているアプリケーションのクラスローダを前記登録手段から削除することを特徴とする請求項16記載のアプリケーション管理方法。

【請求項18】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法であって、前記アプリケーション管理装置は、アプリケーション管理プログラム上で動作するアプリケーションの一つであるアンインストールアプリケーションとして動作し、クライアントからネットワークを介して前記機器にアンインストール指示とアンインストール対象のアプリケーション名が送信されてきた場合、プログラムファイルが保存された外部記憶装置から前記アプリケーション名に対応するプログラムファイルを削除し、アプリケーションのクラスローダが登録される登録手段、及びアプリケーションのプログラムファイルの保存場所が記憶される記憶手段から前記アプリケーション名に対応するエントリを削除することを特徴とするアプリケーション管理方法。

【請求項19】 前記仮想マシンは、Java (Sun Microsystemsの登録商標) 仮想マシンを含み、複数のアプリケーションが動作可能であることを特徴とする請求項11、12、14、15、16、18の何れかに記載のアプリケーション管理方法。

【請求項20】 複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機、デジタルカメラ等の組み込み機器に対するアプリケーションのインストール、起動、停止、アンインストールに適用可能であることを特徴とする請求項11乃至18の何れかに記載のアプリケーション管理方法。

【請求項21】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法を実行するプログラムを記憶したコンピュータにより読み出し可能な記憶媒体であって、前記アプリケーション管理方法は、クライアントからネットワークを介して前記機器にインストール指示とプログラムファイルが送信されてきた場合、前記プログラムファイルを外部記憶装置に保存するステップと、前記プログラムファイルの位置情報を、アプリケーションのプログラムファイルに関する情報が記憶される記憶手段に登録するステップとを有することを特徴とする記憶媒体。

【請求項22】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法を実行するプログラムを記憶したコンピュータにより読み出し可能な記憶媒体であって、前記アプリケーション管理方法は、クライアントからネットワークを介して前記機器にアプリケーションの停止指示があった場合は、前記停止指示があったアプリケーションを停止するステップと、前記クライアントから前記ネットワークを介してアプリケーションの起動指示があった場合で、前記起動指示があったアプリケーション用のクラスローダが、アプリケーションのクラスローダが登録される登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成するステップと、アプリケーションを起動するステップと、前記作成したクラスローダを前記登録手段に登録するステップと、前記作成したクラスローダによりクラスをロードするステップと、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されている場合は、前記登録手段に登録されているクラスローダを取得するステップと、前記取得したクラスローダによりクラスをロードするステップと、アプリケーションを起動するステップとを有することを特徴とする記憶媒体。

【請求項23】 前記アプリケーション管理方法は、アプリケーションの実行に必要なメモリが不足した場合、アプリケーションの実行状態を管理する状態管理手段に停止状態と記憶されているアプリケーションのクラスローダを前記登録手段から削除するステップを有することを特徴とする請求項22記載の記憶媒体。

【請求項24】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置

に適用されるアプリケーション管理方法を実行するプログラムを記憶したコンピュータにより読み出し可能な記憶媒体であって、

前記アプリケーション管理方法は、クライアントからネットワークを介して前記機器にアンインストール指示とアンインストール対象のアプリケーション名が送信されてきた場合、プログラムファイルが保存された外部記憶装置から前記アプリケーション名に対応するプログラムファイルを削除するステップと、アプリケーションのクラスローダが登録される登録手段、及びアプリケーションのプログラムファイルの保存場所が記憶される記憶手段から前記アプリケーション名に対応するエントリを削除するステップとを有することを特徴とする記憶媒体。

【請求項25】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法を実行するプログラムを記憶したコンピュータにより読み出し可能な記憶媒体であって、

前記アプリケーション管理方法は、前記アプリケーション管理装置を、アプリケーション管理プログラム上で動作するアプリケーションの一つであるインストールアプリケーションとして動作させ、クライアントからネットワークを介して前記機器にプログラムファイルが送信されてきた場合、前記プログラムファイルを外部記憶装置に保存するステップと、前記プログラムファイルの位置情報を、アプリケーションのプログラムファイルに関する情報が記憶される記憶手段に登録するステップとを有することを特徴とする記憶媒体。

【請求項26】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法を実行するプログラムを記憶したコンピュータにより読み出し可能な記憶媒体であって、

前記アプリケーション管理方法は、前記アプリケーション管理装置を、アプリケーション管理プログラム上で動作するアプリケーションの一つである起動・停止アプリケーションとして動作させ、クライアントからネットワークを介して前記機器にアプリケーションの停止指示があった場合は、前記停止指示があったアプリケーションを停止するように、アプリケーションの起動・停止を行う起動・停止手段に指示するステップと、前記クライアントから前記ネットワークを介してアプリケーションの起動指示があった場合、前記起動指示があったアプリケーション用のクラスローダが、アプリケーションのクラスローダが登録される登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成するステップと、アプリケーションを起動するステップと、前記作成したクラスローダを前記登録手段に登録するステップと、前記作成したクラスローダによりクラスをロードするステップと、前記起動指示があったアプリケーシ

ョン用のクラスローダが前記登録手段に登録されている場合は、前記登録手段に登録されているクラスローダを取得するステップと、前記取得したクラスローダによりクラスをロードするステップと、アプリケーションを起動するように前記起動・停止手段に指示するステップとを有することを特徴とする記憶媒体。

【請求項27】 前記アプリケーション管理方法は、アプリケーションの実行に必要なメモリが不足した場合、アプリケーションの実行状態を管理する状態管理手段に停止状態と記憶されているアプリケーションのクラスローダを前記登録手段から削除するステップを有することを特徴とする請求項26記載の記憶媒体。

【請求項28】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法を実行するプログラムを記憶したコンピュータにより読み出し可能な記憶媒体であって、

前記アプリケーション管理方法は、前記アプリケーション管理装置を、アプリケーション管理プログラム上で動作するアプリケーションの一つであるアンインストールアプリケーションとして動作させ、クライアントからネットワークを介して前記機器にアンインストール指示とアンインストール対象のアプリケーション名が送信されてきた場合、プログラムファイルが保存された外部記憶装置から前記アプリケーション名に対応するプログラムファイルを削除するステップと、アプリケーションのクラスローダが登録される登録手段、及びアプリケーションのプログラムファイルの保存場所が記憶される記憶手段から前記アプリケーション名に対応するエントリを削除するステップとを有することを特徴とする記憶媒体。

【請求項29】 前記仮想マシンは、Java (Sun Microsystemsの登録商標) 仮想マシンを含み、複数のアプリケーションが動作可能であることを特徴とする請求項21、22、24、25、26、28の何れかに記載の記憶媒体。

【請求項30】 複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機、デジタルカメラ等の組込み機器においてアプリケーションのインストール、起動、停止、アンインストールを行う場合に、前記組込み機器に適用される前記アプリケーション管理方法を実行するプログラムを記憶したコンピュータにより読み出し可能であることを特徴とする請求項21乃至28の何れかに記載の記憶媒体。

【請求項31】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に供給されるプログラムであって、クライアントからネットワークを介して前記機器にインストール指示とプログラムファイルが送信されてきた場合、前記プログラムファイルを外部記憶装置に保存するステップと、前記プログラムファイルの位置情報を、ア

アプリケーションのプログラムファイルに関する情報が記憶される記憶手段に登録するステップとを記憶したことを特徴とするプログラム。

【請求項32】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に供給されるプログラムであって、

クライアントからネットワークを介して前記機器にアプリケーションの停止指示があった場合は、前記停止指示があったアプリケーションを停止するステップと、前記クライアントから前記ネットワークを介してアプリケーションの起動指示があった場合で、前記起動指示があったアプリケーション用のクラスローダが、アプリケーションのクラスローダに登録される登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成するステップと、アプリケーションを起動するステップと、前記作成したクラスローダを前記登録手段に登録するステップと、前記作成したクラスローダによりクラスをロードするステップと、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されている場合は、前記登録手段に登録されているクラスローダを取得するステップと、前記取得したクラスローダによりクラスをロードするステップと、アプリケーションを起動するステップとを記憶したことを特徴とするプログラム。

【請求項33】 アプリケーションの実行に必要なメモリが不足した場合、アプリケーションの実行状態を管理する状態管理手段に停止状態と記憶されているアプリケーションのクラスローダを前記登録手段から削除するステップを記憶したことを特徴とする請求項32記載のプログラム。

【請求項34】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に供給されるプログラムであって、クライアントからネットワークを介して前記機器にアンインストール指示とアンインストール対象のアプリケーション名が送信されてきた場合、プログラムファイルが保存された外部記憶装置から前記アプリケーション名に対応するプログラムファイルを削除するステップと、アプリケーションのクラスローダが登録される登録手段、及びアプリケーションのプログラムファイルの保存場所が記憶される記憶手段から前記アプリケーション名に対応するエントリを削除するステップとを記憶したことを特徴とするプログラム。

【請求項35】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に供給されるプログラムであって、前記アプリケーション管理装置を、アプリケーション管理プログラム上で動作するアプリケーションの一つであるインストールアプリケーションとして動作させ、クライアントからネットワークを介して前記機器にプログラ

ムファイルが送信されてきた場合、前記プログラムファイルを外部記憶装置に保存するステップと、前記プログラムファイルの位置情報を、アプリケーションのプログラムファイルに関する情報が記憶される記憶手段に登録するステップとを記憶したことを特徴とするプログラム。

【請求項36】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に供給されるプログラムであって、

前記アプリケーション管理装置を、アプリケーション管理プログラム上で動作するアプリケーションの一つである起動・停止アプリケーションとして動作させ、クライアントからネットワークを介して前記機器にアプリケーションの停止指示があった場合は、前記停止指示があったアプリケーションを停止するように、アプリケーションの起動・停止を行う起動・停止手段に指示するステップと、前記クライアントから前記ネットワークを介してアプリケーションの起動指示があった場合で、前記起動指示があったアプリケーション用のクラスローダが、アプリケーションのクラスローダに登録される登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成するステップと、アプリケーションを起動するステップと、前記作成したクラスローダを前記登録手段に登録するステップと、前記作成したクラスローダによりクラスをロードするステップと、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されている場合は、前記登録手段に登録されているクラスローダを取得するステップと、前記取得したクラスローダによりクラスをロードするステップと、アプリケーションを起動するように前記起動・停止手段に指示するステップとを記憶したことを特徴とするプログラム。

【請求項37】 アプリケーションの実行に必要なメモリが不足した場合、アプリケーションの実行状態を管理する状態管理手段に停止状態と記憶されているアプリケーションのクラスローダを前記登録手段から削除するステップを記憶したことを特徴とする請求項36記載のプログラム。

【請求項38】 機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に供給されるプログラムであって、前記アプリケーション管理装置を、アプリケーション管理プログラム上で動作するアプリケーションの一つであるアンインストールアプリケーションとして動作させ、クライアントからネットワークを介して前記機器にアンインストール指示とアンインストール対象のアプリケーション名が送信されてきた場合、プログラムファイルが保存された外部記憶装置から前記アプリケーション名に対応するプログラムファイルを削除するステップと、アプリケーションのクラスローダが登録される登録手段、

及びアプリケーションのプログラムファイルの保存場所が記憶される記憶手段から前記アプリケーション名に対応するエントリを削除するステップとを記憶したことを特徴とするプログラム。

【請求項39】 前記仮想マシンは、Java (Sun Microsystemsの登録商標) 仮想マシンを含み、複数のアプリケーションが動作可能であることを特徴とする請求項31、32、34、35、36、38の何れかに記載のプログラム。

【請求項40】 複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機、デジタルカメラ等の組込み機器においてアプリケーションのインストール、起動、停止、アンインストールを行う場合に、前記組込み機器に供給されることを特徴とする請求項31乃至38の何れかに記載のプログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、アプリケーション管理装置、アプリケーション管理方法、記憶媒体及びプログラムに関し、特に、複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機(MFP)、デジタルカメラ等の組込み機器上でアプリケーションプログラムを実行する場合に適用して好適なアプリケーション管理装置、アプリケーション管理方法、記憶媒体及びプログラムに関する。

【0002】

【従来の技術】 近年、組込み機器に搭載されるプロセッサ、メモリ、表示装置は、高機能化すると共に大容量化が進展している。また、組込み機器では、例えば複写機においても、単純にコピーに使用するだけでなく、ネットワークに接続することでプリンタやFAXとしても使用することが可能な複合機(Multi Function Printer: MFP)が主流となりつつある。

【0003】 上述したような組込み機器は、ある程度の解像度をもった液晶ディスプレイなどの表示装置を装備しており、ユーザが各種機能を使用するためのアプリケーションプログラムを実行することが可能な構成となっている。しかし、そのようなアプリケーションプログラムはROMに固定的に書き込まれるため、ユーザがアプリケーションプログラムを簡単に交換したり変更したりすることは不可能であった。

【0004】 例えば特開2000-250758号公報に開示されているように、ヒューレットパッカード社が組込み機器用Java (Sun Microsystemsの登録商標) アプリケーションマネージャの特許を出願している。同公報記載の技術では、動作させるJavaプログラムのクラス(Javaにおいてプログラムの一つの単位)をメモリ上に展開し管理を行うクラスローダは、Javaプログラムごとに用意されていないため、ユーザやサードパーティが作成したJavaプログラムの中に同一の名前をもった異なる

クラスが存在した場合、Java仮想マシンは、どのクラスを実行したらよいか判断することができず、複数のJavaプログラムの実行ができなくなってしまうという面がある。

【0005】

【発明が解決しようとする課題】 しかしながら、上記従来技術においては下記のような問題があった。即ち、上述したように、組込み機器のアプリケーションプログラムはROMに固定的に書き込まれるため、ユーザがアプリケーションプログラムを簡単に交換したり変更したりすることは不可能であり、障害が発生した場合も、サービスマンなどによりROMを交換するなどの方法によってしか、アプリケーションプログラムを修正することができなかった。

【0006】 また、ユーザが複数の組込み機器を使用する場合、特定のユーザ用のアプリケーションプログラムをインストール可能な組込み機器のところまで行かなければ、作業を行うことができず、管理が困難で煩雑であるという問題があった。

【0007】 また、従来の組込み機器のアプリケーションプログラムは、組込み機器の起動時に立ち上げられていたが、アプリケーションプログラムをインストール可能な組込み機器においては、インストールされたアプリケーションプログラムの起動・停止などができることが必要となる。何故ならば、そのようなアプリケーションプログラムは、組込み機器の資源の問題もあり、常に動作している必要はなく、ある特定の条件やユーザが使用する場合にのみ必要とされるものであることが多いからである。

【0008】 一方、組込み機器のようなリソースの限られた機器において、複数のJava仮想マシンを動作させることは困難である。何故ならば、Java仮想マシンは高速なプロセッサと大量のメモリを必要とするが、多くの組込み機器はそれほど高速なプロセッサや大量のメモリを装備していないからである。

【0009】 また、組み込み機器を制御するオペレーティングシステムによっては、同じプログラムを複数個実行することができなかったり、更には一度起動したプログラムを停止できなかったり、一度停止すると再起動ができないようなものも存在する。このような機器の場合、一つのJavaプログラムを立ち上げる度に一つのJava仮想マシンが起動し、起動したJavaプログラムを使う必要がなくなっても、機器の電源を落とすまで不要なJava仮想マシンが動作し続けるということになってしまう。

【0010】 そのため、一つのVM(仮想マシン)で複数のJavaアプリケーションプログラムを動作させるためのアプリケーション管理手段が必要となる。そして、アプリケーションプログラムに、一つのVM上で複数のJavaアプリケーションプログラムを動作させている場合、あるJavaアプリケーションプログラムは、動作には影響を与

えずに、他のJavaアプリケーションプログラムのインストール、起動、停止、アンインストールなどを行う必要がでてくる。

【0011】また、組込み機器にプログラムが容易にインストールできるようになると、ユーザやサードパーティが作成したJavaプログラムのクラス名の衝突が起こることが予想されるという問題が発生する。

【0012】本発明は、上述した点に鑑みなされたものであり、複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機(MFP)、デジタルカメラ等の組込み機器を出荷した後に、組込み機器にプログラムをインストールしたり、更にインストールしたプログラムを起動したりすることを可能としたアプリケーション管理装置、アプリケーション管理方法、記憶媒体及びプログラムを提供することを第一の目的とする。

【0013】また、本発明は、上述した点に鑑みなされたものであり、使用していないプログラムのクラスローダをメモリから削除することにより、組込み機器のメモリを節約することを可能としたアプリケーション管理装置、アプリケーション管理方法、記憶媒体及びプログラムを提供することを第二の目的とする。

【0014】また、本発明は、上述した点に鑑みなされたものであり、インストール、起動・停止、アンインストールを行うプログラム自体のインストール、起動・停止、アンインストールも簡単に行うことを可能としたアプリケーション管理装置、アプリケーション管理方法、記憶媒体及びプログラムを提供することを第三の目的とする。

【0015】

【課題を解決するための手段】上記目的を達成するため、請求項1記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置であって、アプリケーションのプログラムファイルに関する情報を記憶する記憶手段と、クライアントと通信を行うネットワーク接続手段と、前記クライアントから前記ネットワーク接続手段を介してインストール指示とプログラムファイルが送信されてきた場合、前記プログラムファイルを外部記憶装置に保存し、前記プログラムファイルの位置情報を前記記憶手段に登録する登録制御手段とを有することを特徴とする。

【0016】上記目的を達成するため、請求項2記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置であって、アプリケーションのクラスローダを登録する登録手段と、アプリケーションのプログラムファイルの保存場所を記憶する記憶手段と、クライアントと通信を行うネットワーク接続手段と、前記クライアントから前記ネットワーク接続手段を介してアプリケーションの停止指示があった場合は、前記停止指示があったアプリケーション

を停止し、前記クライアントから前記ネットワーク接続手段を介してアプリケーションの起動指示があった場合、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成し、アプリケーションを起動し、前記作成したクラスローダを前記登録手段に登録し、前記作成したクラスローダによりクラスをロードし、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されている場合は、前記登録手段に登録されているクラスローダを取得し、前記取得したクラスローダによりクラスをロードし、アプリケーションを起動する起動・停止手段とを有することを特徴とする。

【0017】上記目的を達成するため、請求項3記載の発明は、アプリケーションの実行状態を管理する状態管理手段を有し、アプリケーションの実行に必要なメモリが不足した場合、前記状態管理手段に停止状態と記憶されているアプリケーションのクラスローダを前記登録手段から削除することを特徴とする。

【0018】上記目的を達成するため、請求項4記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置であって、アプリケーションのクラスローダを登録する登録手段と、アプリケーションのプログラムファイルの保存場所を記憶する記憶手段と、クライアントと通信を行うネットワーク接続手段と、前記クライアントから前記ネットワーク接続手段を介してアンインストール指示とアンインストール対象のアプリケーション名が送信されてきた場合、プログラムファイルが保存された外部記憶装置から前記アプリケーション名に対応するプログラムファイルを削除し、前記登録手段及び前記記憶手段から前記アプリケーション名に対応するエントリを削除する削除制御手段とを有することを特徴とする。

【0019】上記目的を達成するため、請求項5記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置であって、アプリケーションのプログラムファイルに関する情報を記憶する記憶手段と、クライアントと通信を行うネットワーク接続手段と、アプリケーション管理装置上で動作するアプリケーションの一つであるインストールアプリケーションとして動作し、前記クライアントから前記ネットワーク接続手段を介してプログラムファイルが送信されてきた場合、前記プログラムファイルを外部記憶装置に保存し、前記プログラムファイルの位置情報を前記記憶手段に登録する登録制御手段とを有することを特徴とする。

【0020】上記目的を達成するため、請求項6記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置であって、アプリケーションの起動・停止を行う起動・停止手

段と、アプリケーションのクラスローダを登録する登録手段と、アプリケーションのプログラムファイルの保存場所を記憶する記憶手段と、クライアントと通信を行うネットワーク接続手段と、アプリケーション管理装置上で動作するアプリケーションの一つである起動・停止アプリケーションとして動作し、前記クライアントから前記ネットワーク接続手段を介してアプリケーションの停止指示があった場合は、前記停止指示があったアプリケーションを停止するように前記起動・停止手段に指示し、前記クライアントから前記ネットワーク接続手段を介してアプリケーションの起動指示があった場合で、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成し、アプリケーションを起動し、前記作成したクラスローダを前記登録手段に登録し、前記作成したクラスローダによりクラスをロードし、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されている場合は、前記登録手段に登録されているクラスローダを取得し、前記取得したクラスローダによりクラスをロードし、アプリケーションを起動するように前記起動・停止手段に指示する起動・停止制御手段とを有することを特徴とする。

【0021】上記目的を達成するため、請求項7記載の発明は、アプリケーションの実行状態を管理する状態管理手段を有し、アプリケーションの実行に必要なメモリが不足した場合、前記状態管理手段に停止状態と記憶されているアプリケーションのクラスローダを前記登録手段から削除することを特徴とする。

【0022】上記目的を達成するため、請求項8記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置であって、アプリケーションの起動・停止を行う起動・停止手段と、アプリケーションのクラスローダを登録する登録手段と、アプリケーションのプログラムファイルの保存場所を記憶する記憶手段と、クライアントと通信を行うネットワーク接続手段と、アプリケーション管理装置上で動作するアプリケーションの一つであるアンインストールアプリケーションとして動作し、前記クライアントから前記ネットワーク接続手段を介してアンインストール指示とアンインストール対象のアプリケーション名が送信されてきた場合、プログラムファイルが保存された外部記憶装置から前記アプリケーション名に対応するプログラムファイルを削除し、前記登録手段及び前記記憶手段から前記アプリケーション名に対応するエントリを削除する削除制御手段とを有することを特徴とする。

【0023】上記目的を達成するため、請求項9記載の発明は、前記仮想マシンは、Java (Sun Microsystemsの登録商標) 仮想マシンを含み、複数のアプリケーションが動作可能であることを特徴とする。

【0024】上記目的を達成するため、請求項10記載

の発明は、複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機、デジタルカメラ等の組込み機器に対するアプリケーションのインストール、起動、停止、アンインストールに適用可能であることを特徴とする。

【0025】上記目的を達成するため、請求項11記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法であって、クライアントからネットワークを介して前記機器にインストール指示とプログラムファイルが送信されてきた場合、前記プログラムファイルを外部記憶装置に保存し、前記プログラムファイルの位置情報を、アプリケーションのプログラムファイルに関する情報が記憶される記憶手段に登録することを特徴とする。

【0026】上記目的を達成するため、請求項12記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法であって、クライアントからネットワークを介して前記機器にアプリケーションの停止指示があった場合は、前記停止指示があったアプリケーションを停止し、前記クライアントから前記ネットワークを介してアプリケーションの起動指示があった場合で、前記起動指示があったアプリケーション用のクラスローダが、アプリケーションのクラスローダに登録される登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成し、アプリケーションを起動し、前記作成したクラスローダを前記登録手段に登録し、前記作成したクラスローダによりクラスをロードし、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されている場合は、前記登録手段に登録されているクラスローダを取得し、前記取得したクラスローダによりクラスをロードし、アプリケーションを起動することを特徴とする。

【0027】上記目的を達成するため、請求項13記載の発明は、アプリケーションの実行に必要なメモリが不足した場合、アプリケーションの実行状態を管理する状態管理手段に停止状態と記憶されているアプリケーションのクラスローダを前記登録手段から削除することを特徴とする。

【0028】上記目的を達成するため、請求項14記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法であって、クライアントからネットワークを介して前記機器にアンインストール指示とアンインストール対象のアプリケーション名が送信されてきた場合、プログラムファイルが保存された外部記憶装置から前記アプリケーション名に対応するプログラムファイルを削除し、アプリケーションのクラスローダが登録される登録手段、及びアプリケーションの

プログラムファイルの保存場所が記憶される記憶手段から前記アプリケーション名に対応するエントリを削除することを特徴とする。

【0029】上記目的を達成するため、請求項15記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法であって、前記アプリケーション管理装置は、アプリケーション管理プログラム上で動作するアプリケーションの一つであるインストールアプリケーションとして動作し、クライアントからネットワークを介して前記機器にプログラムファイルが送信されてきた場合、前記プログラムファイルを外部記憶装置に保存し、前記プログラムファイルの位置情報を、アプリケーションのプログラムファイルに関する情報が記憶される記憶手段に登録することを特徴とする。

【0030】上記目的を達成するため、請求項16記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法であって、前記アプリケーション管理装置は、アプリケーション管理プログラム上で動作するアプリケーションの一つである起動・停止アプリケーションとして動作し、クライアントからネットワークを介して前記機器にアプリケーションの停止指示があった場合は、前記停止指示があったアプリケーションを停止するように、アプリケーションの起動・停止を行う起動・停止手段に指示し、前記クライアントから前記ネットワークを介してアプリケーションの起動指示があった場合で、前記起動指示があったアプリケーション用のクラスローダが、アプリケーションのクラスローダが登録される登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成し、アプリケーションを起動し、前記作成したクラスローダを前記登録手段に登録し、前記作成したクラスローダによりクラスをロードし、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されている場合は、前記登録手段に登録されているクラスローダを取得し、前記取得したクラスローダによりクラスをロードし、アプリケーションを起動するように前記起動・停止手段に指示することを特徴とする。

【0031】上記目的を達成するため、請求項17記載の発明は、アプリケーションの実行に必要なメモリが不足した場合、アプリケーションの実行状態を管理する状態管理手段に停止状態と記憶されているアプリケーションのクラスローダを前記登録手段から削除することを特徴とする。

【0032】上記目的を達成するため、請求項18記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法であって、前記アプリケーション管理装置は、アプリケーション管理プログラ

ム上で動作するアプリケーションの一つであるアンインストールアプリケーションとして動作し、クライアントからネットワークを介して前記機器にアンインストール指示とアンインストール対象のアプリケーション名が送信されてきた場合、プログラムファイルが保存された外部記憶装置から前記アプリケーション名に対応するプログラムファイルを削除し、アプリケーションのクラスローダが登録される登録手段、及びアプリケーションのプログラムファイルの保存場所が記憶される記憶手段から前記アプリケーション名に対応するエントリを削除することを特徴とする。

【0033】上記目的を達成するため、請求項19記載の発明は、前記仮想マシンは、Java (Sun Microsystemsの登録商標) 仮想マシンを含み、複数のアプリケーションが動作可能であることを特徴とする。

【0034】上記目的を達成するため、請求項20記載の発明は、複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機、デジタルカメラ等の組込み機器に対するアプリケーションのインストール、起動、停止、アンインストールに適用可能であることを特徴とする。

【0035】上記目的を達成するため、請求項21記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法を実行するプログラムを記憶したコンピュータにより読み出し可能な記憶媒体であって、前記アプリケーション管理方法は、クライアントからネットワークを介して前記機器にインストール指示とプログラムファイルが送信されてきた場合、前記プログラムファイルを外部記憶装置に保存するステップと、前記プログラムファイルの位置情報を、アプリケーションのプログラムファイルに関する情報が記憶される記憶手段に登録するステップとを有することを特徴とする。

【0036】上記目的を達成するため、請求項22記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法を実行するプログラムを記憶したコンピュータにより読み出し可能な記憶媒体であって、前記アプリケーション管理方法は、クライアントからネットワークを介して前記機器にアプリケーションの停止指示があった場合は、前記停止指示があったアプリケーションを停止するステップと、前記クライアントから前記ネットワークを介してアプリケーションの起動指示があった場合で、前記起動指示があったアプリケーション用のクラスローダが、アプリケーションのクラスローダが登録される登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成するステップと、アプリケーションを起動するステップと、前記作成したクラスローダを前記登録手段に登録するステ

ップと、前記作成したクラスローダによりクラスをロードするステップと、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されている場合は、前記登録手段に登録されているクラスローダを取得するステップと、前記取得したクラスローダによりクラスをロードするステップと、アプリケーションを起動するステップとを有することを特徴とする。

【0037】上記目的を達成するため、請求項23記載の発明は、前記アプリケーション管理方法は、アプリケーションの実行に必要なメモリが不足した場合、アプリケーションの実行状態を管理する状態管理手段に停止状態と記憶されているアプリケーションのクラスローダを前記登録手段から削除するステップを有することを特徴とする。

【0038】上記目的を達成するため、請求項24記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法を実行するプログラムを記憶したコンピュータにより読み出し可能な記憶媒体であって、前記アプリケーション管理方法は、クライアントからネットワークを介して前記機器にアンインストール指示とアンインストール対象のアプリケーション名が送信されてきた場合、プログラムファイルが保存された外部記憶装置から前記アプリケーション名に対応するプログラムファイルを削除するステップと、アプリケーションのクラスローダが登録される登録手段、及びアプリケーションのプログラムファイルの保存場所が記憶される記憶手段から前記アプリケーション名に対応するエントリを削除するステップとを有することを特徴とする。

【0039】上記目的を達成するため、請求項25記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法を実行するプログラムを記憶したコンピュータにより読み出し可能な記憶媒体であって、前記アプリケーション管理方法は、前記アプリケーション管理装置を、アプリケーション管理プログラム上で動作するアプリケーションの一つであるインストールアプリケーションとして動作させ、クライアントからネットワークを介して前記機器にプログラムファイルが送信されてきた場合、前記プログラムファイルを外部記憶装置に保存するステップと、前記プログラムファイルの位置情報を、アプリケーションのプログラムファイルに関する情報が記憶される記憶手段に登録するステップとを有することを特徴とする。

【0040】上記目的を達成するため、請求項26記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法を実行するプログラムを記憶したコンピュータにより読み出し可能な記憶媒体

であって、前記アプリケーション管理方法は、前記アプリケーション管理装置を、アプリケーション管理プログラム上で動作するアプリケーションの一つである起動・停止アプリケーションとして動作させ、クライアントからネットワークを介して前記機器にアプリケーションの停止指示があった場合は、前記停止指示があったアプリケーションを停止するように、アプリケーションの起動・停止を行う起動・停止手段に指示するステップと、前記クライアントから前記ネットワークを介してアプリケーションの起動指示があった場合で、前記起動指示があったアプリケーション用のクラスローダが、アプリケーションのクラスローダが登録される登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成するステップと、アプリケーションを起動するステップと、前記作成したクラスローダを前記登録手段に登録するステップと、前記作成したクラスローダによりクラスをロードするステップと、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されている場合は、前記登録手段に登録されているクラスローダを取得するステップと、前記取得したクラスローダによりクラスをロードするステップと、アプリケーションを起動するように前記起動・停止手段に指示するステップとを有することを特徴とする。

【0041】上記目的を達成するため、請求項27記載の発明は、前記アプリケーション管理方法は、アプリケーションの実行に必要なメモリが不足した場合、アプリケーションの実行状態を管理する状態管理手段に停止状態と記憶されているアプリケーションのクラスローダを前記登録手段から削除するステップを有することを特徴とする。

【0042】上記目的を達成するため、請求項28記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に適用されるアプリケーション管理方法を実行するプログラムを記憶したコンピュータにより読み出し可能な記憶媒体であって、前記アプリケーション管理方法は、前記アプリケーション管理装置を、アプリケーション管理プログラム上で動作するアプリケーションの一つであるアンインストールアプリケーションとして動作させ、クライアントからネットワークを介して前記機器にアンインストール指示とアンインストール対象のアプリケーション名が送信されてきた場合、プログラムファイルが保存された外部記憶装置から前記アプリケーション名に対応するプログラムファイルを削除するステップと、アプリケーションのクラスローダが登録される登録手段、及びアプリケーションのプログラムファイルの保存場所が記憶される記憶手段から前記アプリケーション名に対応するエントリを削除するステップとを有することを特徴とする。

【0043】上記目的を達成するため、請求項29記載

の発明は、前記仮想マシンは、Java (Sun Microsystems の登録商標) 仮想マシンを含み、複数のアプリケーションが動作可能であることを特徴とする。

【0044】上記目的を達成するため、請求項30記載の発明は、複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機、デジタルカメラ等の組込み機器においてアプリケーションのインストール、起動、停止、アンインストールを行う場合に、前記組込み機器に適用される前記アプリケーション管理方法を実行するプログラムを記憶したコンピュータにより読み出し可能であることを特徴とする。

【0045】上記目的を達成するため、請求項31記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に供給されるプログラムであって、クライアントからネットワークを介して前記機器にインストール指示とプログラムファイルが送信されてきた場合、前記プログラムファイルを外部記憶装置に保存するステップと、前記プログラムファイルの位置情報を、アプリケーションのプログラムファイルに関する情報が記憶される記憶手段に登録するステップとを記憶したことを特徴とする。

【0046】上記目的を達成するため、請求項32記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に供給されるプログラムであって、クライアントからネットワークを介して前記機器にアプリケーションの停止指示があった場合は、前記停止指示があったアプリケーションを停止するステップと、前記クライアントから前記ネットワークを介してアプリケーションの起動指示があった場合で、前記起動指示があったアプリケーション用のクラスローダが、アプリケーションのクラスローダが登録される登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成するステップと、アプリケーションを起動するステップと、前記作成したクラスローダを前記登録手段に登録するステップと、前記作成したクラスローダによりクラスをロードするステップと、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されている場合は、前記登録手段に登録されているクラスローダを取得するステップと、前記取得したクラスローダによりクラスをロードするステップと、アプリケーションを起動するステップとを記憶したことを特徴とする。

【0047】上記目的を達成するため、請求項33記載の発明は、アプリケーションの実行に必要なメモリが不足した場合、アプリケーションの実行状態を管理する状態管理手段に停止状態と記憶されているアプリケーションのクラスローダを前記登録手段から削除するステップを記憶したことを特徴とする。

【0048】上記目的を達成するため、請求項34記載の発明は、機器における仮想マシン上で動作するアプリ

ケーションを管理するアプリケーション管理装置に供給されるプログラムであって、クライアントからネットワークを介して前記機器にアンインストール指示とアンインストール対象のアプリケーション名が送信されてきた場合、プログラムファイルが保存された外部記憶装置から前記アプリケーション名に対応するプログラムファイルを削除するステップと、アプリケーションのクラスローダが登録される登録手段、及びアプリケーションのプログラムファイルの保存場所が記憶される記憶手段から前記アプリケーション名に対応するエントリを削除するステップとを記憶したことを特徴とする。

【0049】上記目的を達成するため、請求項35記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に供給されるプログラムであって、前記アプリケーション管理装置を、アプリケーション管理プログラム上で動作するアプリケーションの一つであるインストールアプリケーションとして動作させ、クライアントからネットワークを介して前記機器にプログラムファイルが送信されてきた場合、前記プログラムファイルを外部記憶装置に保存するステップと、前記プログラムファイルの位置情報を、アプリケーションのプログラムファイルに関する情報が記憶される記憶手段に登録するステップとを記憶したことを特徴とする。

【0050】上記目的を達成するため、請求項36記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に供給されるプログラムであって、前記アプリケーション管理装置を、アプリケーション管理プログラム上で動作するアプリケーションの一つである起動・停止アプリケーションとして動作させ、クライアントからネットワークを介して前記機器にアプリケーションの停止指示があった場合は、前記停止指示があったアプリケーションを停止するように、アプリケーションの起動・停止を行う起動・停止手段に指示するステップと、前記クライアントから前記ネットワークを介してアプリケーションの起動指示があった場合で、前記起動指示があったアプリケーション用のクラスローダが、アプリケーションのクラスローダが登録される登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成するステップと、アプリケーションを起動するステップと、前記作成したクラスローダを前記登録手段に登録するステップと、前記作成したクラスローダによりクラスをロードするステップと、前記起動指示があったアプリケーション用のクラスローダが前記登録手段に登録されている場合は、前記登録手段に登録されているクラスローダを取得するステップと、前記取得したクラスローダによりクラスをロードするステップと、アプリケーションを起動するように前記起動・停止手段に指示するステップとを記憶したことを特徴とする。

【0051】上記目的を達成するため、請求項37記載の発明は、アプリケーションの実行に必要なメモリが不足した場合、アプリケーションの実行状態を管理する状態管理手段に停止状態と記憶されているアプリケーションのクラスローダを前記登録手段から削除するステップを記憶したことを特徴とする。

【0052】上記目的を達成するため、請求項38記載の発明は、機器における仮想マシン上で動作するアプリケーションを管理するアプリケーション管理装置に供給されるプログラムであって、前記アプリケーション管理装置を、アプリケーション管理プログラム上で動作するアプリケーションの一つであるアンインストールアプリケーションとして動作させ、クライアントからネットワークを介して前記機器にアンインストール指示とアンインストール対象のアプリケーション名が送信されてきた場合、プログラムファイルが保存された外部記憶装置から前記アプリケーション名に対応するプログラムファイルを削除するステップと、アプリケーションのクラスローダが登録される登録手段、及びアプリケーションのプログラムファイルの保存場所が記憶される記憶手段から前記アプリケーション名に対応するエントリを削除するステップとを記憶したことを特徴とする。

【0053】上記目的を達成するため、請求項39記載の発明は、前記仮想マシンは、Java（Sun Microsystemsの登録商標）仮想マシンを含み、複数のアプリケーションが動作可能であることを特徴とする。

【0054】上記目的を達成するため、請求項40記載の発明は、複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機、デジタルカメラ等の組込み機器においてアプリケーションのインストール、起動、停止、アンインストールを行う場合に、前記組込み機器に供給されることを特徴とする。

【0055】

【発明の実施の形態】 先ず、本発明の実施の形態を説明する前に、本発明の概要について説明する。

【0056】本発明で用いるJava仮想マシンは、Sun Microsystemsが開発したJava言語によって記述されたプログラムを実行する仮想マシンである。Java仮想マシン上で動作する複数のJavaアプリケーション（以後Serviceと記述する）の動作を管理するアプリケーション管理プログラム（以後ServiceManagerと記述する）と、そのアプリケーション管理プログラム上で動作するWebサーバであるHTTPサーバServiceと、そのHTTPサーバService上で動作するJavaプログラムの一形式であるServletとして動作するInstallServlet及びControlServletを用いることにより、遠隔地のWebブラウザなどから機器にServiceをインストールしたり、起動・停止を行うことで、組込み機器のような機器において実行されている他のJavaアプリケーションの実行に影響を与えずに、機器の出荷後に簡単にプログラムを追加したり、動作させたり、

削除したりすることを可能とするものである。

【0057】更に、個々のServiceのプログラムデータの単位であるクラスデータをディスクなどから読み込んだり、読み込んだクラスデータを管理するためにJavaに用意されているクラスローダを、各々のService毎に生成し管理することにより、Serviceのクラス名の衝突を回避したり、クラスローダごとメモリから削除することにより、空きメモリを増やすなどといったことを簡単に実現するものである。

【0058】以下、本発明の第1の実施の形態乃至第4の実施の形態を図面に基づいて詳細に説明する。

【0059】〔第1の実施の形態〕図1は本発明の第1の実施の形態に係るアプリケーション管理システムの構成を示すブロック図である。アプリケーション管理システムは、Java仮想マシン107、Javaアプリケーション（以後Serviceと記述する）のインストールを行うユーザ側に設置されたWebブラウザ100、Serviceの管理を行うServiceManager101、ServiceManager101上のServiceとして実行されるHTTP（HyperText Transfer Protocol）サーバ102、HTTPサーバ102上で実行され且つHTTPサーバ102上で動作するJavaプログラムの一形式であるServletの形式で作成され、Serviceのインストール処理を行うInstallServlet103、インストールされるプログラムが保存される外部記憶装置105から構成されている。図中106a及び106bはServiceである。尚、図中104は第2の実施の形態の構成要素である。

【0060】図2は本発明の第1の実施の形態に係るServiceManager101の構成を示すブロック図である。ServiceManager101は、インストールされているServiceの名称と、Serviceのプログラムファイルが保存されているパス情報の対を記憶するクラスパス情報管理部202から構成されている。尚、図中201、203は第2の実施の形態の構成要素である。

【0061】図3は本発明の第1の実施の形態に係るクラスパス情報管理部202の構成を示す説明図である。クラスパス情報管理部202には、上述したようにServiceの名称とパス情報の対が記憶される。

【0062】次に、上記の如く構成された本発明の第1の実施の形態の動作を図1、図5、図11を参照しながら詳細に説明する。

【0063】先ず、図5に示すように、ServiceManager101上で動作するServiceの一つであるHTTPサーバ102上で動作しServiceのインストールを行うInstallServlet103は、Webブラウザ100からの要求に基づき、図11に示すように、Service名称111を入力する欄1101、ServiceのプログラムデータのWebブラウザ側でのパスを入力する欄1102、転送ボタン1103から構成されるフォームデータを送信する（ステップS500）。

【0064】ユーザがWebブラウザ100において、前記フォームにService名称111、Webブラウザ側でのパスを入力し、転送ボタン1103を押すと、HTML（HyperText Markup Language）で規定されたフォームの仕様に従い、HTTPプロトコルでプログラムのインストール指示110として、インストールすべきServiceの名称111、Serviceのプログラムデータ112が、Webブラウザ100からInstallServlet103に転送されてくる（ステップS501）。

【0065】InstallServlet103は、Webブラウザ100から送られてきたプログラムデータ112を受け取ると（ステップS502）、プログラムデータ112を外部記憶装置105にファイルとして保存し（ステップS503）、更にプログラムを保存したパスとService名称111をクラスパス管理部202に登録する（ステップS504）。

【0066】以上説明したように、本発明の第1の実施の形態によれば、Java仮想マシン107上で動作する複数のJavaアプリケーション（Service）の動作を管理するアプリケーション管理プログラムであるServiceManager101と、ServiceManager101上で動作するWebサーバであるHTTPサーバ102と、HTTPサーバ102上で動作するJavaプログラムの一形式であるServletとして動作するInstallServlet103を備えることにより、遠隔地のWebブラウザ100などから機器にServiceをインストールしたり、起動・停止を行うことで、組込み機器のような機器において実行されている他のJavaアプリケーションの実行に影響を与えずに、機器の出荷後に簡単にプログラムを追加したり、動作させたり、削除したりすることが可能となる。

【0067】更に、個々のServiceのプログラムデータの単位であるクラスデータをディスクなどから読み込んだり、読み込んだクラスデータを管理するためにJavaに用意されているクラスローダを、各々のService毎に生成し管理することにより、Serviceのクラス名の衝突を回避したり、クラスローダごとメモリから削除することにより、空きメモリを増やすなどといったことを簡単に実現することができる。

【0068】即ち、複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機（MFP）、デジタルカメラ等の組込み機器を出荷した後に、組込み機器にプログラムをインストールしたり、更にインストールしたプログラムを起動したりすることが可能となる。また、使用していないプログラムのクラスローダをメモリから削除することにより、組込み機器のメモリを節約することが可能となる。更に、インストール、起動・停止、アンインストールを行うプログラムは、アプリケーション管理プログラム上で動くプログラムであるため、インストール、起動・停止、アンインストールを行うプログラム自体のインストール、起動・停止、ア

ンインストールも簡単に行うことが可能となる。

【0069】〔第2の実施の形態〕図1は本発明の第2の実施の形態に係るアプリケーション管理システムの構成を示すブロック図である。アプリケーション管理システムは、Java仮想マシン107、Javaアプリケーション（以後Serviceと記述する）のインストールを行うユーザ側に設置されたWebブラウザ100、Serviceの管理を行うServiceManger101、ServiceManager101上のServiceとして実行されるHTTPサーバ102、HTTPサーバ102上で実行され且つHTTPサーバ102上で動作するJavaプログラムの一形式であるServletの形式で作成され、Serviceの起動・停止処理を行うControlServlet104、インストールされているプログラムが保存される外部記憶装置105から構成されている。図中106a及び106bはServiceである。尚、図中103は上記第1の実施の形態の構成要素である。

【0070】図2は本発明の第2の実施の形態に係るServiceManager101の構成を示すブロック図である。ServiceManager101は、Serviceの起動・停止を行うService起動部201、インストールされているServiceの名称と、Serviceのプログラムファイルが保存されているパス情報の対を記憶するクラスパス情報管理部202、Serviceの名称と、Serviceが起動したときにService起動部201で生成される前記起動されたServiceのクラスローダの対を記憶するクラスローダ管理部203から構成されている。

【0071】図3は本発明の第2の実施の形態に係るクラスパス情報管理部202の構成を示す説明図、図4は本発明の第2の実施の形態に係るクラスローダ管理部203の構成を示す説明図である。クラスパス情報管理部202には、上述したようにServiceの名称とパス情報の対が記憶され、クラスローダ管理部203には、上述したようにServiceの名称とクラスローダの対が記憶される。

【0072】次に、上記の如く構成された本発明の第2の実施の形態の動作を図1、図6、図12を参照しながら詳細に説明する。

【0073】先ず、図6に示すように、ServiceManager101上で動作するServiceの一つであるHTTPサーバ102上で動作しServiceのインストールを行うControlServlet104は、Webブラウザ100からの要求に基づき、図12に示すように、Service名とチェックボックスからなるサービス名リスト1201、起動ボタン1202、停止ボタン1203から構成されるフォームデータを送信する（ステップS600）。

【0074】ユーザがWebブラウザ100において、前記フォームのサービス名リスト1201で起動したいServiceのチェックボックスをチェックし、起動ボタン1202を押すと、HTMLで規定されたフォームの仕様に従い、HTTPプロトコルでプログラムの起動指示113とし

て、起動すべきServiceの名称114が、Webブラウザ100からControlServlet104に転送されてくる（ステップS601）。

【0075】更に、Webブラウザ100からプログラムの起動指示113として、起動すべきServiceの名称114がControlServlet104に転送されてくると（ステップS601）、ControlServlet104は、Webブラウザ100から送られてきた起動すべきServiceの名称114を受け取り（ステップS602）、ServiceManager101に対して起動指示を行う。

【0076】ServiceManager101のService起動部201は、ControlServlet104から起動指示を受け取ると、先ず、クラスパス情報管理部202にService名称114が登録されているか否かのチェックを行う（ステップS603）。クラスパス情報管理部202に対象Serviceが存在しなければ、処理を終了する。クラスパス情報管理部202に対象Serviceが存在すれば、次に、対象Serviceが停止しているかどうかのチェックを行う（ステップS604）。対象Serviceが停止していなければ、処理を終了する。対象Serviceが停止していたら、クラスロード管理部203に対象Serviceを実行するのに必要な対象Serviceのクラスロードが登録されているか否かをチェックする（ステップS605）。

【0077】クラスロード管理部203に対象Serviceを実行するのに必要な対象Serviceのクラスロードが登録されていなければ、対象Service用のクラスロードを生成する。生成されたクラスロードは、クラスパス情報管理部202からプログラムデータの保管場所を取得し、プログラムデータのファイルを外部記憶装置105からロードし、Serviceを起動し（ステップS607）、クラスロード管理部203に上記ステップS607で生成したクラスロードを登録する（ステップS608）。クラスロード管理部203に対象Serviceを実行するのに必要な対象Serviceのクラスロードが登録されていた場合には、クラスロードから対象Serviceのクラスロードを取得し、Serviceを起動する（ステップS606）。

【0078】以上説明したように、本発明の第2の実施の形態によれば、Java仮想マシン107上で動作する複数のJavaアプリケーション（Service）の動作を管理するアプリケーション管理プログラムであるServiceManager101と、ServiceManager101上で動作するWebサーバであるHTTPサーバ102と、HTTPサーバ102上で動作するJavaプログラムの一形式であるServletとして動作するControlServlet104を備えることにより、遠隔地のWebブラウザ100などから機器にServiceをインストールしたり、起動・停止を行うことで、組込み機器のような機器において実行されている他のJavaアプリケーションの実行に影響を与えずに、機器の出荷後に簡単にプログラムを追加したり、動作させたり、削除したり

することが可能となる。

【0079】更に、個々のServiceのプログラムデータの単位であるクラスデータをディスクなどから読み込んだり、読み込んだクラスデータを管理するためにJavaに用意されているクラスローダを、各々のService毎に生成し管理することにより、Serviceのクラス名の衝突を回避したり、クラスローダごとメモリから削除することにより、空きメモリを増やすなどといったことを簡単に実現することができる。

【0080】即ち、複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機（MFP）、デジタルカメラ等の組込み機器を出荷した後に、組込み機器にプログラムをインストールしたり、更にインストールしたプログラムを起動したりすることが可能となり、また、組込み機器のメモリを節約することが可能となり、また、インストール、起動・停止、アンインストールを行うプログラム自体のインストール、起動・停止、アンインストールも簡単にを行うことが可能となる。

【0081】〔第3の実施の形態〕図7は本発明の第3の実施の形態に係るアプリケーション管理システムの構成を示すブロック図である。アプリケーション管理システムは、Java仮想マシン107、Javaアプリケーション（以後Serviceと記述する）のインストールを行うユーザ側に設置されたWebブラウザ100、Serviceの管理を行うServiceManager101、ServiceManager101上のServiceとして実行されるHTTPサーバ102、HTTPサーバ102上で実行され且つHTTPサーバ102上で動作するJavaプログラムの一形式であるServletの形式で作成され、Serviceのアンインストール処理を行うUninstallServlet701、インストールされているプログラムが保存される外部記憶装置105から構成されている。

【0082】更に、ServiceManager101は、Serviceのアンインストール処理を行うサービスアンインストール部702、インストールされているServiceの名称とServiceのプログラムファイルが保存されているパス情報の対を記憶するクラスパス情報管理部202、Serviceの名称とServiceが起動したときにService起動部で生成される前記起動されたServiceのクラスロードの対を記憶するクラスロード管理部203から構成されている。

【0083】次に、上記の如く構成された本発明の第3の実施の形態の動作を図7、図8、図13を参照しながら詳細に説明する。

【0084】先ず、図8に示すように、ServiceManager101上で動作するServiceの一つであるHTTPサーバ102上で動作しServiceのアンインストール処理を行うUninstallServlet701は、Webブラウザ100からの要求に基づき、図13に示すように、Service名とチェックボックスからなるサービス名リスト1301、アンインストールボタン1302から構成されるフォームデー

タを送信する(ステップS800)。

【0085】ユーザがWebブラウザ100において、前記フォームのサービス名リスト1301でアンインストールしたいServiceのチェックボックスをチェックし、アンインストールボタン1302を押すと、HTMLで規定されたフォームの仕様に従い、HTTPプロトコルでプログラムのアンインストール指示710として、アンインストールすべきServiceの名称711が、Webブラウザ100からUninstallServlet701に転送されてくる(ステップS801)。

【0086】Webブラウザ100からプログラムのアンインストール指示710として、アンインストールすべきServiceの名称711が転送されてくると(ステップS801)、UninstallServlet701は、Webブラウザ100から送られてきたアンインストール指示710とアンインストールすべきServiceの名称711を受け取り(ステップS802)、ServiceManager101に対してServiceのアンインストール指示712を行う。

【0087】ServiceManager101のサービスアンインストール部702は、UninstallServlet701からアンインストール指示を受け取ると、まず、クラスパス情報管理部202にService名称711が登録されているかどうかのチェックを行う(ステップS803)。クラスパス情報管理部202に対象Serviceが存在しなければ、処理を終了する。クラスパス情報管理部202に対象Serviceが存在すれば、次に、対象Serviceが停止しているかどうかのチェックを行う(ステップS804)。

【0088】対象Serviceが停止していなければ、処理を終了する。対象Serviceが停止していたら、クラスローダ管理部203から対象Serviceのクラスローダ削除714を行う(ステップS805)。更に、クラスパス管理部202から対象Serviceのクラスパス情報削除713を行った後(ステップS806)、外部記憶装置105から対象Serviceのプログラムファイル削除715を行い(ステップS807)、処理を終了する。

【0089】以上説明したように、本発明の第3の実施の形態によれば、上記第1の実施の形態と同様に、複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機(MFP)、デジタルカメラ等の組込み機器を出荷した後に、組込み機器にプログラムをインストールしたり、更にインストールしたプログラムを起動したりすることが可能となり、また、組込み機器のメモリを節約することが可能となり、また、インストール、起動・停止、アンインストールを行うプログラム自体のインストール、起動・停止、アンインストールも簡単に行うことが可能となる。

【0090】[第4の実施の形態] 図9は本発明の第4の実施の形態に係るアプリケーション管理システムの構成を示すブロック図である。アプリケーション管理システムは、Java仮想マシン900、Javaアプリケーション

(以後Serviceと記述する)管理を行うServiceManager101から構成されている。

【0091】更に、ServiceManager101は、Serviceのメモリ管理を行うServiceメモリ管理部902、インストールされているServiceの名称とServiceの実行状態を管理するService状態管理部901、Serviceの名称とServiceが起動したときにService起動部で生成される前記起動されたServiceのクラスローダの対を記憶するクラスローダ管理部203から構成されている。

【0092】次に、上記の如く構成された本発明の第4の実施の形態の動作を図9、図10を参照しながら詳細に説明する。

【0093】Javaプログラムの実行中にメモリが不足すると、図10に示すように、Java仮想マシン900から、ServiceManager101のServiceメモリ管理部902にメモリの開放要求910が送られる(ステップS1001)。Serviceメモリ管理部902は、Service状態管理部901から現在動作していないService名を取得する(ステップS1002)。Service状態管理部901は、停止状態のServiceがあるか否かを調べる(ステップS1003)。

【0094】停止状態のServiceがなければ、処理を終了する。停止状態のServiceがあれば、そのクラスローダ管理部203から該当Serviceのクラスローダを削除し(ステップS1004でYes)、処理を終了する。クラスローダ管理部203から該当Serviceのクラスローダを削除できなければ(ステップS1004でNo)、上記ステップS1002の処理に戻る。

【0095】以上説明したように、本発明の第4の実施の形態によれば、上記第1の実施の形態と同様に、複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機(MFP)、デジタルカメラ等の組込み機器を出荷した後に、組込み機器にプログラムをインストールしたり、更にインストールしたプログラムを起動したりすることが可能となり、また、組込み機器のメモリを節約することが可能となり、また、インストール、起動・停止、アンインストールを行うプログラム自体のインストール、起動・停止、アンインストールも簡単に行うことが可能となる。

【0096】上述した本発明の実施形態では、アプリケーション管理システムが適用される機器については特定しなかったが、複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機(MFP)、デジタルカメラ等の各種の組込み機器に適用することができる。

【0097】尚、本発明は、複数の機器から構成されるシステムに適用しても、1つの機器からなる装置に適用してもよい。上述した実施形態の機能を実現するソフトウェアのプログラムコードを記憶した記憶媒体等の媒体をシステム或いは装置に供給し、そのシステム或いは装

置のコンピュータ（またはCPUやMPU）が記憶媒体等の媒体に格納されたプログラムコードを読み出し実行することによっても、達成されることは言うまでもない。

【0098】この場合、記憶媒体等の媒体から読み出されたプログラムコード自体が上述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体等の媒体は本発明を構成することになる。プログラムコードを供給するための記憶媒体等の媒体としては、例えば、フロッピー（登録商標）ディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、CD-R、磁気テープ、不揮発性のメモリカード、ROM、或いはネットワークを介したダウンロードなどを用いることができる。

【0099】また、コンピュータが読み出したプログラムコードを実行することにより、上述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼動しているOSなどが実際の処理の一部または全部を行い、その処理によって上述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0100】更に、記憶媒体等の媒体から読み出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書き込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によって上述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0101】

【発明の効果】以上説明したように、本発明のアプリケーション管理装置によれば、クライアントからインストール指示とプログラムファイルが送信されてきた場合、プログラムファイルを外部記憶装置に保存し、プログラムファイルの位置情報を記憶手段に登録する制御を行う。

【0102】また、クライアントからアプリケーションの停止指示があった場合は、停止指示があったアプリケーションを停止し、クライアントからアプリケーションの起動指示があった場合で、起動指示があったアプリケーション用のクラスローダが登録手段に登録されていない場合は、アプリケーション用のクラスローダを作成し、アプリケーションを起動し、作成したクラスローダを登録手段に登録し、作成したクラスローダによりクラスをロードし、起動指示があったアプリケーション用のクラスローダが登録手段に登録されている場合は、登録されているクラスローダを取得し、取得したクラスローダによりクラスをロードし、アプリケーションを起動する制御を行う。

【0103】また、クライアントからアンインストール

指示とアンインストール対象のアプリケーション名が送信されてきた場合、外部記憶装置からアプリケーション名に対応するプログラムファイルを削除し、登録手段及び記憶手段からアプリケーション名に対応するエントリを削除する制御を行う。

【0104】これにより、複写機、プリンタ、コピー機能・プリンタ機能・ファクシミリ機能を有する複合機（MFP）、デジタルカメラ等の組込み機器を出荷した後に、組込み機器にプログラムをインストールしたり、更にインストールしたプログラムを起動したりすることが可能となる。また、使用していないプログラムのクラスローダをメモリから削除することにより、組込み機器のメモリを節約することが可能となる。更に、インストール、起動・停止、アンインストールを行うプログラムは、アプリケーション管理プログラム上で動くプログラムであるため、インストール、起動・停止、アンインストールを行うプログラム自体のインストール、起動・停止、アンインストールも簡単に行うことが可能となる。

【0105】また、本発明のアプリケーション管理方法、本発明の記憶媒体、本発明のプログラムにおいても、上記と同様に、組込み機器を出荷した後に、組込み機器にプログラムをインストールしたり、更にインストールしたプログラムを起動したりすることが可能となるなどの効果を奏する。

【図面の簡単な説明】

【図1】本発明の第1、第2の実施の形態に係るアプリケーション管理システムの構成を示すブロック図である。

【図2】第1、第2の実施の形態に係るServiceManagerの構成を示すブロック図である。

【図3】第1、第2の実施の形態に係るクラスパス情報管理部の構成を示す説明図である。

【図4】第2の実施の形態に係るクラスローダ管理部の構成を示す説明図である。

【図5】第1の実施の形態に係るインストール処理を示すフローチャートである。

【図6】第2の実施の形態に係る起動処理を示すフローチャートである。

【図7】第3の実施の形態に係るアプリケーション管理システムの構成を示すブロック図である。

【図8】第3の実施の形態に係るアンインストール処理を示すフローチャートである。

【図9】第4の実施の形態に係るアプリケーション管理システムの構成を示すブロック図である。

【図10】第4の実施の形態に係るメモリ管理処理を示すフローチャートである。

【図11】第1の実施の形態に係るServiceインストール画面を示す説明図である。

【図12】第2の実施の形態に係るService起動・停止画面を示す説明図である。

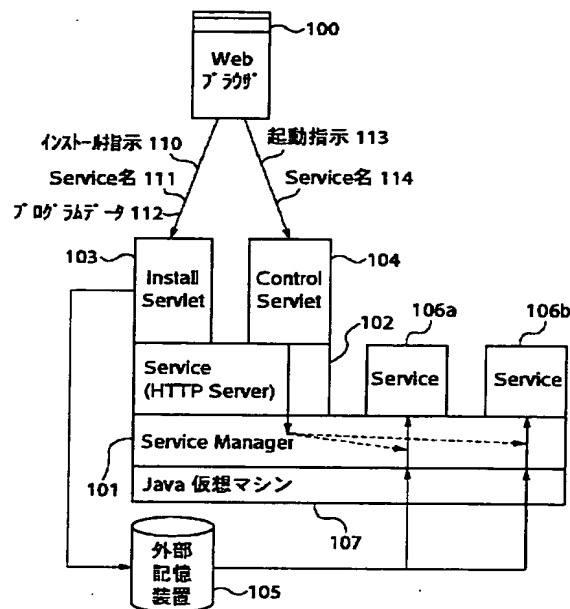
【図13】第3の実施の形態に係るServiceアンインストール画面を示す説明図である。

【符号の説明】

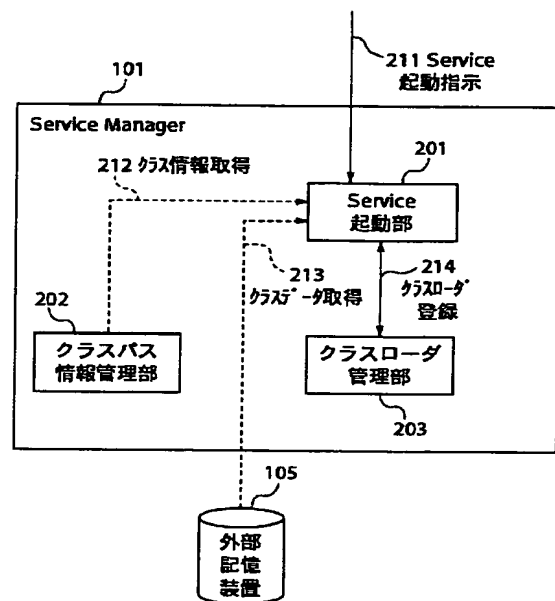
- 100 Webブラウザ
- 101 ServiceManger
- 102 HTTPサーバ
- 103 InstallServlet (ネットワーク接続手段、登録制御手段)
- 104 ControlServlet
- 105 外部記憶装置

- 107 Java仮想マシン
- 201 Service起動部 (起動・停止手段)
- 202 クラスパス情報管理部 (記憶手段)
- 203 クラスローダ管理部 (登録手段)
- 701 UnInstallServlet (ネットワーク接続手段)
- 702 サービスアンインストール部 (削除制御手段)
- 900 Java仮想マシン
- 901 Service状態管理部 (状態管理手段)
- 902 Serviceメモリ管理部

【図1】



【図2】



【図3】

202 クラスパス情報管理部

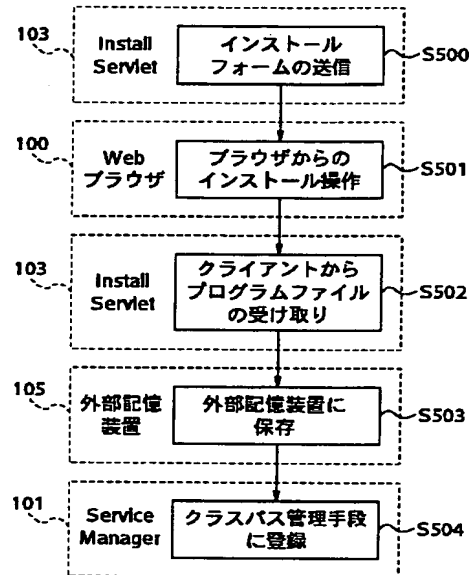
Service A	Applets/ServiceA/a.jar
Service B	Applets/ServiceB/b.jar
Service C	Applets/ServiceC/c.jar
⋮	⋮
⋮	⋮

【図4】

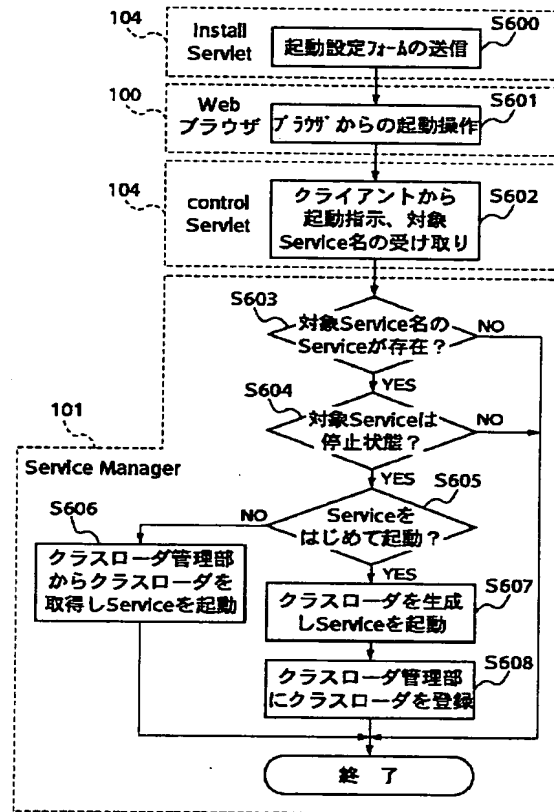
203 クラスローダ管理部

Service A	Service Aのクラスローダオブジェクト
Service B	Service Bのクラスローダオブジェクト
Service C	Service Cのクラスローダオブジェクト
⋮	⋮
⋮	⋮

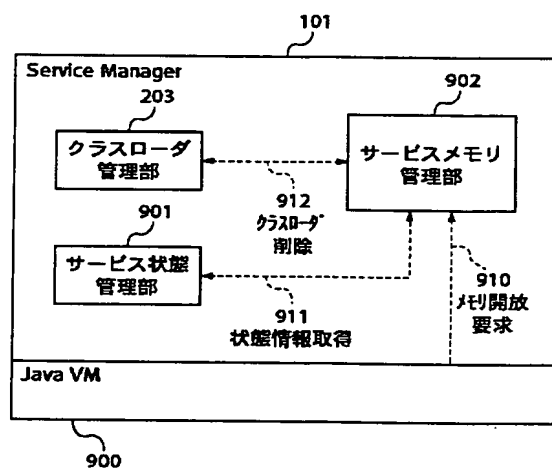
【図5】



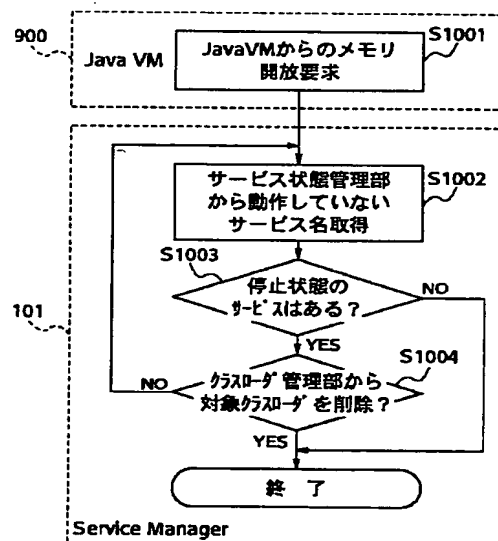
【図6】



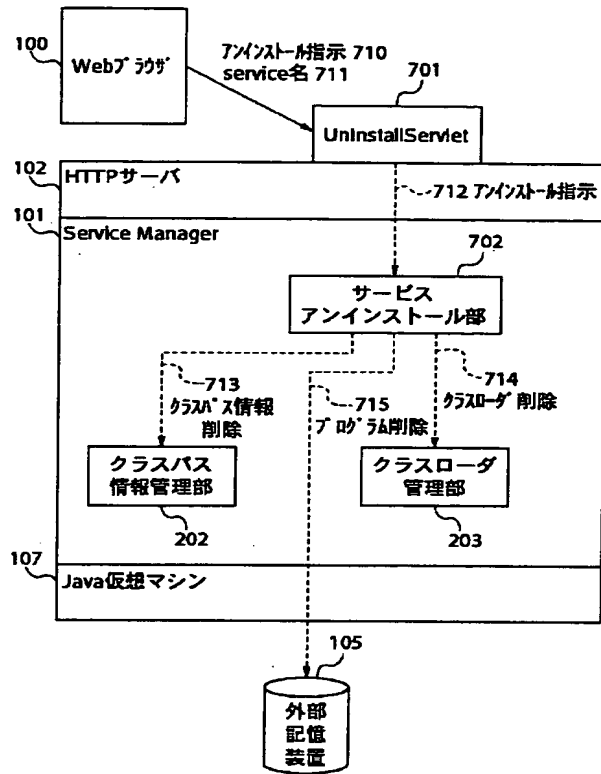
【図9】



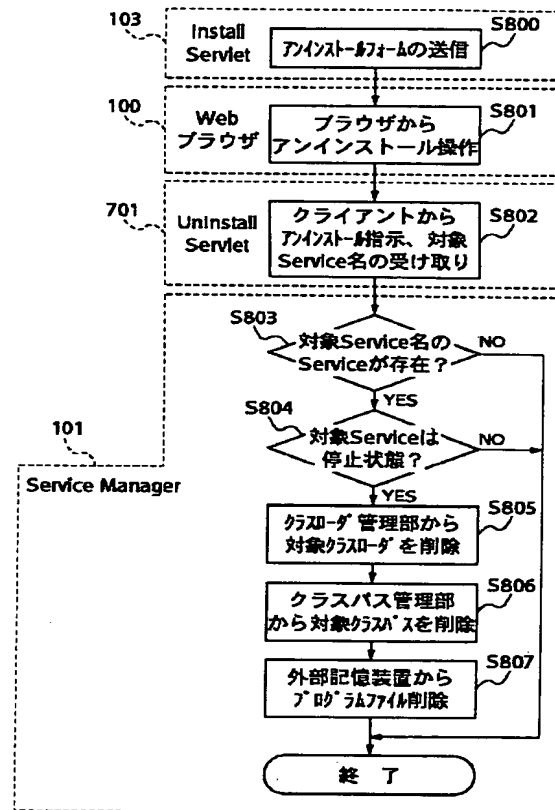
【図10】



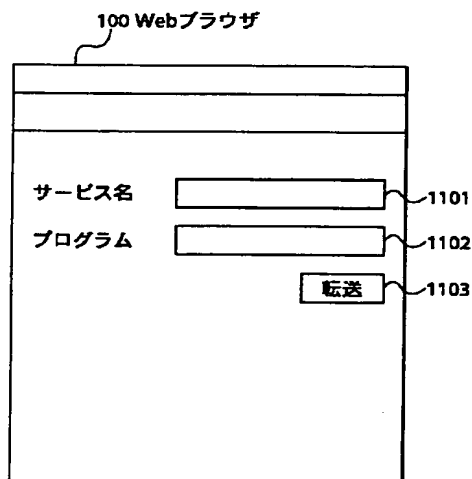
【図7】



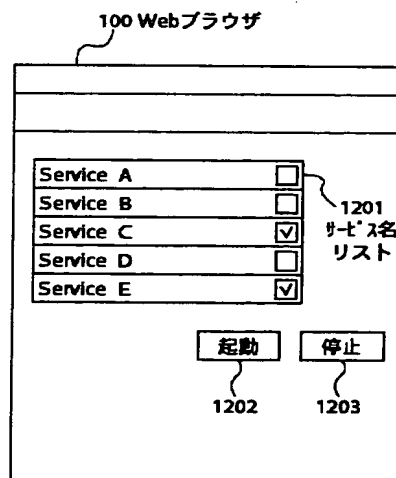
【図8】



【図11】



【図12】



【図13】

